IFW

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/886,665	06/21/2001	Guillaume Comeau	1054.1US	4921

7590 01/05/2005

ZUCOTTO WIRELESS
16644 WEST BERNARDO DRIVE
SUITE 301
SAN DIEGO, CA 92127

EXAMINER	
ZHEN, LI B	
ART UNIT	**PAPER NUMBER**
2126	

DATE MAILED: 01/05/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/886,665	COMEAU ET AL.	
	Examiner	Art Unit	
	Li B. Zhen	2126	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *12 August 2002*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Notice of References Cited	Application/Control No. 09/886,665	Applicant(s)/Patent Under Reexamination COMEAU ET AL.	
	Examiner Li B. Zhen	Art Unit 2126	Page 1 of 1

U.S. PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Name	Classification
	A	US-6,349,312 B1	02-2002	Fresko et al.	707/205
	B	US-6,317,872 B1	11-2001	Gee et al.	717/152
	C	US-6,111,894 A	08-2000	Bender et al.	370/469
	D	US-			
	E	US-			
	F	US-			
	G	US-			
	H	US-			
	I	US-			
	J	US-			
	K	US-			
	L	US-			
	M	US-			

FOREIGN PATENT DOCUMENTS

*		Document Number Country Code-Number-Kind Code	Date MM-YYYY	Country	Name	Classification
	N					
	O					
	P					
	Q					
	R					
	S					
	T					

NON-PATENT DOCUMENTS

*		Include as applicable: Author, Title Date, Publisher, Edition or Volume, Pertinent Pages)
	U	
	V	
	W	
	X	

*A copy of this reference is not being furnished with this Office action. (See MPEP § 707.05(a).)
Dates in MM-YYYY format are publication dates. Classifications may be US or foreign.

DETAILED ACTION

1. Claims 1 – 30 are pending in the application.

Double Patenting

2. A rejection based on double patenting of the "same invention" type finds its support in the language of 35 U.S.C. 101 which states that "whoever invents or discovers any new and useful process ... may obtain a patent therefor ..." (Emphasis added). Thus, the term "same invention," in this context, means an invention drawn to identical subject matter. See *Miller v. Eagle Mfg. Co.*, 151 U.S. 186 (1894); *In re Ockert*, 245 F.2d 467, 114 USPQ 330 (CCPA 1957); and *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970).

A statutory type (35 U.S.C. 101) double patenting rejection can be overcome by canceling or amending the conflicting claims so they are no longer coextensive in scope. The filing of a terminal disclaimer cannot overcome a double patenting rejection based upon 35 U.S.C. 101.

3. Claims 1 – 30 are provisionally rejected under 35 U.S.C. 101 as claiming the same invention as that of claims 1 – 30 of copending Application No. 09/871481. This is a provisional double patenting rejection since the conflicting claims have not in fact been patented.

Claim Rejections - 35 USC § 102

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. **Claims 1 – 7, 11, 14, 21, 22 and 24 are rejected under 35 U.S.C. 102(e) as being anticipated by U.S. Patent NO. 6,349,312 to Fresko.**

5. As to claim 1, Fresko teaches an apparatus for utilizing information, comprising:

a memory, the memory comprising at least one data structure [memory heap 402, Fig. 4; col. 6 line 50 – 54]; and

a plurality of layers, each layer comprising at least one thread [managing multiple threads of execution; col. 6 line 55 – 67], each thread utilizing each data structure from the same portion of the memory [multiple threads are sharing the same pool 408... multiple threads are concurrently requesting memory allocations from the same preallocated pool 408; col. 6, lines 50 – 67].

6. As to claim 21, Fresko teaches utilizing a stream of information in a data path, comprising:

a memory, the memory comprising at least one data structure [memory heap 402, Fig. 4; col. 6 line 50 – 54], each data structure comprising a pointer [PreallocationContext object 112 comprises a start pointer, a current pointer, and an end pointer; col. 4, lines 7 – 23];

a plurality of layers, the data path comprising the plurality of layers [multiple threads are concurrently requesting memory allocations from the same preallocated pool 408; col. 6, lines 50 – 67; processor 704 coupled with bus 702 for processing information; col. 10, lines 25 – 67], the stream of information comprising at least one data structure [communication interface 718 sends and receives electrical, electromagnetic or optical signals that carry digital data streams representing various types of information; col. 11, lines 42 – 56], each layer utilizing each data structure via

its pointer [multiple threads are sharing the same pool 408... multiple threads are concurrently requesting memory allocations from the same preallocated pool 408; col. 6, lines 50 – 67].

7. As to claim 2, Fresko an application layer [thread objects; col. 8, lines 1 – 23] and a hardware layer [processor 704 coupled with bus 702 for processing information; col. 10, lines 25 – 67], wherein the application layer comprises one of the plurality of layers, wherein the hardware layer comprises one of the plurality of layers, wherein the application layer and hardware layer utilize each data structure from the same portion of memory [multiple threads are concurrently requesting memory allocations from the same preallocated pool 408; col. 6, lines 50 – 67].

8. As to claim 3, Fresko teaches wherein at least one of the plurality of layers comprises a realtime thread [memory allocation mechanism will still be deterministic and predictable. It is this determinism that is usually required in systems such as real time systems; col. 9 line 13 – 40].

9. As to claim 4, Fresko teaches wherein each data structure comprises a block object [PreallocationContext object], wherein at least a portion of each block object is comprised of a contiguous portion of the memory [PreallocationContext object 112 is instantiated, the constructor of the object 112 causes a contiguous set of memory space 108 having a size N; col. 3 line 58 – 67].

10. As to claim 5, Fresko teaches wherein the contiguous portion of the memory is defined a byte array [constructor of the object 112 causes a contiguous set of memory space 108 having a size N; col. 3 line 58 – 67].

11. As to claim 6, this is rejected for similar reasons as claim 4 above.

12. As to claim 7, Fresko teaches the apparatus of claim 1, further comprising a Java or Java-like virtual machine [manager 410 takes the form of a Java VM; col. 7, lines 1 – 7], wherein each thread comprises a Java or Java-like thread, wherein the Java or Java-like thread utilizes the same portion of memory independent of Java or Java-like monitors [DoPreallocated method determines (204, FIG. 2) which thread invoked it...DoPreallocated method instantiates (208) the PreallocationContext object 112; col. 5 line 5 – 15].

13. As to claim 11, Fresko teaches information is received by the apparatus as streamed information [communication interface 718 sends and receives electrical, electromagnetic or optical signals that carry digital data streams representing various types of information; col. 11, lines 42 – 56] wherein each data structure is preallocated to the memory prior reception of the information [DoPreallocated method determines (204, FIG. 2) which thread invoked it...DoPreallocated method instantiates (208) the PreallocationContext object 112; col. 5 line 5 – 15].

14. As to claim 14, Fresko teaches a virtual machine [manager 410 takes the form of a Java VM; col. 7, lines 1 – 7] utilizing a garbage collection mechanism, the virtual machine running each thread, each thread utilizing the same portion of the memory independent of the garbage collection mechanism [trigger a GC operation if the free space on the heap is below a certain threshold; col. 3 line 26 – 57].

15. As to claims 22 and 24, these are rejected for similar reasons as claims 2 and 14 above.

Claim Rejections - 35 USC § 103

16. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

17. **Claims 12, 13, and 15 – 20 are rejected under 3.5 U.S.C. 103(a) as being unpatentable over Fresko in view of U.S. Patent NO. 6,317,872 to Gee.**

18. As to claim 15, Fresko does not specifically teach the garbage collection mechanism comprises a thread, Java-like threads, each thread comprising a priority, and the priority of the Java-like threads is higher than the priority of the garbage collection thread.

However, Gee teaches a garbage collection mechanism comprising a thread [garbage collection is accommodated by an indirect referencing technique which can be bypassed for objects necessary for real time operation; col. 3, line 55 – col. 4, line 12], Java-like threads, each thread comprising a priority [JVM is actually composed of one or more threads. Each JVM thread has a private JAVA stack; col. 6, lines 45 – 55], and the priority of the Java-like threads is higher than the priority of the garbage collection thread [garbage collection on garbage-collected objects may be interrupted by some real time event which is handled using non-garbage-collected objects as the processing and storage mechanism; col. 19, line 9-21].

19. It would have been obvious to a person of ordinarily skilled in the art at the time the invention to apply the teaching of maintaining the priority of the Java-like threads to be higher than the priority of the garbage collection thread as taught by Gee to the invention of Fresko because this insures that access to non-garbage-collected objects, such as real-time control objects, is never delayed by the unavailability of the processor or impeded by garbage collection; col. 19, lines 10 – 22 of Gee].

20. As to claims 12 and 16, Fresko as modified teaches each data structure comprises a block object [PreallocationContext object 112; col. 3 line 58 – 67 of Fresko], and further comprising a freelist data structure and at least one queue data structure [data structure, referred to hereinafter as the "ready queue", used to represent all ready threads should facilitate fast lookup of the highest priority thread; col. 28, lines 1 – 10 of Gee], each block object comprising a respective handle, wherein at any given

time the respective handle belongs to the freelist data structure or a queue data structure [ready queue is based on an array of linked lists of differing priorities... Each priority level has a doubly linked list of thread control blocks, with each linked list referred to as a priority queue. There is a global variable for each list and the variables reside in a pointer array; col. 28, lines 1 – 40 of Gee].

21. As to claim 13, Fresko as modified teaches a protocol stack [JEM processor is the use of six stack registers (S0-S5) in on-board register file 204. Together, these registers function as a "cache" memory and act as an extension of the accumulator stack which would normally be located in external data memory 106; col. 9, lines 52 – 65 of Gee], the protocol stack residing in the memory, wherein the protocol stack preallocates each block to the freelist data structure [data structure, referred to hereinafter as the "ready queue", used to represent all ready threads should facilitate fast lookup of the highest priority thread; col. 28, lines 1 – 10 of Gee].

22. As to claim 17, Fresko as modified teaches at least one queue data structure [data structure, referred to hereinafter as the "ready queue", used to represent all ready threads should facilitate fast lookup of the highest priority thread; col. 28, lines 1 – 10 of Gee]; at least one frame data structure, each frame data structure comprising an instance of one or more block objects [PreallocationContext object 112; col. 3 line 58 – 67 of Fresko], each block object comprising a respective handle, each queue data structure capable of holding an instance of at least one frame data structure [Each JVM

thread has a private JAVA stack, created at the same time as the thread, which stores JVM frames; col. 6, lines 45 – 55 of Gee], and each thread using the queue data structure to pass a block handle to another thread [ready queue is based on an array of linked lists of differing priorities...Each priority level has a doubly linked list of thread control blocks, with each linked list referred to as a priority queue. There is a global variable for each list and the variables reside in a pointer array; col. 28, lines 1 – 40 of Gee].

23. As to claim 18, Fresko as modified teaches a virtual machine running each thread [manager 410 takes the form of a Java VM; col. 7, lines 1 – 7 of Fresko], at least one queueendpoint comprising at least one of the threads [are executed or "played" in accordance with their position in a ready queue; col. 27, lines 1 – 15 of Gee], at least one queue [control structures] comprising ends bounded by a queueendpoint, each queue for holding each of data structures in a data path for use by each qaeuendpoint [JEM microcode directly manipulates the ready queue and other threading control structures, in a manner determined by a predetermined priority scheme. The ECB fields shown include fields which facilitate this piano roll and priority mechanism; col. 27, lines 1 – 30 of Gee], each queue notifies a respective queueendpoint when the queue needs to be serviced by the queueendpoint [RqPtrArray comprised of fields Rq0Ptr 1625 through Rq32Ptr 1627 is used as the "ready queue". It is organized as an array of doubly-linked TCB lists, where each array element corresponds to a priority level; col. 27, lines 10 – 67 of Gee], a queueendpoint passes instances of each data structure

from one queue to another queue by a respective handle belonging to the data structure [array provides for rapid queue insertion an deletion and for thread dispatch prioritization; col. 28, lines 1 – 60 of Gee].

24. As to claim 19, Fresko as modified teaches a queue notifies a respective queueendpoint upon the occurrence of a queue empty event, queue not empty event, queue congested event, or queue not congested event [individual linked lists are known as "priority queues" and the global variables (1754, 1758) reside in the ready queue 1750 which is effectively a pointer array. There are 32 JEM priority levels, running from 0 to 31. Empty priority levels, such as level 1766, have null pointers in the corresponding array entries; col. 29 line 31-45 of Gee].

25. As to claim 20, Fresko as modified teaches a queue status data structure shared by a queue and a respective queueendpoint [JEM microcode directly manipulates the ready queue and other threading control structures, in a manner determined by a predetermined priority scheme. The ECB fields shown include fields which facilitate this piano roll and priority mechanism; col. 27, lines 1 – 30 of Gee], wherein the queue sets a flag in the data status structure to notify the respective queueendpoint when the queue needs to be serviced [The JCB includes event flags, validity and error information; col. 25 line 57 – col. 26, line 10 of Gee].

26. **Claims 8 – 10, 23 and 25 – 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Fresko in view of U.S. Patent NO. 6,111,894 to Bender.**

27. As to claim 25, Fresko teaches the invention substantially as claimed including a system for utilizing data structure with a plurality of threads, comprising;

a memory, the memory comprising at least one data structure [memory heap 402, Fig. 4; col. 6 line 50 – 54]; and

a plurality of threads, the plurality of threads utilizing the data structures [multiple threads are sharing the same pool 408... multiple threads are concurrently requesting memory allocations from the same preallocated pool 408; col. 6, lines 50 – 67].

28. Although Fresko teaches the invention substantially as claimed, Fresko does not specifically teach an interrupt mechanism for enabling and disabling interrupts.

However, Bender teaches a hardware abstraction layer [col. 1, lines 50 – 65] and an interrupt mechanism for enabling and disabling interrupts [default mode for both the send and receive interfaces is polling mode (interrupts disabled); col. 25, lines 54 – 67].

29. It would have been obvious to a person of ordinarily skilled in the art at the time of the invention to apply the teaching of an interrupt mechanism for enabling and disabling interrupts as taught by Bender to the invention of Fresko because this provides powerful flexibility to any job scheduler and management system [col. 2, lines 1 – 15 of Bender].

30. As to claim 27, Fresko as modified teaches a system for accessing streaming information with a plurality of threads, comprising:

a memory [memory heap 402, Fig. 4; col. 6 line 50 – 54 of Fresko]; and

interrupt means for enabling and disabling interrupts [default mode for both the send and receive interfaces is polling mode (interrupts disabled); col. 25, lines 54 – 67 of Bender];

wherein the plurality of threads [multiple threads are sharing the same pool 408... multiple threads are concurrently requesting memory allocations from the same preallocated pool 408; col. 6, lines 50 – 67 of Fresko] access the streaming information from the memory [communication interface 718 sends and receives electrical, electromagnetic or optical signals that carry digital data streams representing various types of information; col. 11, lines 42 – 56 of Fresko] by disabling the interrupts via the interrupt means [polling mode (interrupts disabled); col. 25, lines 54 – 67 of Bender].

31. As to claim 29, Fresko as modified teaches a method for accessing information in a memory with a plurality of threads, comprising the steps of:

transferring information from one thread to another thread via handles to the information [multiple threads are sharing the same pool 408... multiple threads are concurrently requesting memory allocations from the same preallocated pool 408; col. 6, lines 50 – 67 of Fresko]; and

disabling interrupts via the threads before performing the step of transferring the information [default mode for both the send and receive interfaces is polling mode (interrupts disabled); col. 25, lines 54 – 67 of Bender].

32. As to claim 8, Fresko as modified teaches interrupt means for disabling interrupts [default mode for both the send and receive interfaces is polling mode (interrupts disabled); col. 25, lines 54 – 67 of Bender], and a Java or Java-like virtual machine capable of executing each thread [manager 410 takes the form of a Java VM; col. 7, lines 1 – 7 of Fresko], wherein each thread utilizes the same portion of memory after the interrupts are disabled by the interrupt means [DoPreallocated method determines (204, FIG. 2) which thread invoked it...DoPreallocated method instantiates (208) the PreallocationContext object 112; col. 5 line 5 – 15 of Fresko].

33. As to claim 9, Fresko as modified teaches wherein all interrupts are disabled [default mode for both the send and receive interfaces is polling mode (interrupts disabled); col. 25, lines 54 – 67 of Bender] before each thread utilizes the same portion of memory [DoPreallocated method determines (204, FIG. 2) which thread invoked it...DoPreallocated method instantiates (208) the PreallocationContext object 112; col. 5 line 5 – 15 of Fresko].

34. As to claim 10, Fresko as modified teaches the threads disable the interrupts via the interrupt means [default mode for both the send and receive interfaces is polling mode (interrupts disabled); col. 25, lines 54 – 67 of Bender].

35. As to claim 23, Fresko as modified teaches interrupt disabling mechanism [default mode for both the send and receive interfaces is polling mode (interrupts disabled); col. 25, lines 54 – 67 of Bender]; and at least one queue, each queue disposed in the data path between a first layer and a second layer, the first layer comprising a producer thread [multiple threads are sharing the same pool 408... multiple threads are concurrently requesting memory allocations from the same preallocated pool 408; col. 6, lines 50 – 67 of Fresko], the second layer comprising a consumer thread, the producer thread for enqueuing each data structure onto a queue, the consumer thread for dequeing each data structure from the queue [communication interface 718 sends and receives electrical, electromagnetic or optical signals that carry digital data streams representing various types of information; col. 11, lines 42 – 56 of Fresko], wherein prior to dequeing and enqueing each data structure interrupts are disabled [polling mode (interrupts disabled); col. 25, lines 54 – 67 of Bender].

36. As to claim 26, 28, 30 these are rejected for similar reasons as claim 2 above.

Conclusion

37. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Li B. Zhen whose telephone number is (703) 305-3406. The examiner can normally be reached on Mon - Fri, 8:30am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (703) 305-9678. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Li B. Zhen
Examiner
Art Unit 2126

lbz
May 25, 2004

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

Organization TC2100 Bldg./Room PK2
U. S. DEPARTMENT OF COMMERCE
COMMISSIONER FOR PATENTS
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450
IF UNDELIVERABLE RETURN IN TEN DAYS

OFFICIAL BUSINESS

U.S. OFFICIAL MAIL
PENALTY FOR
PRIVATE USE $300

UNITED STATES POSTAGE
PITNEY BOWES

02 1A $ 01.06⁰
0004204034 FEB 17 2005
MAILED FROM ZIPCODE 22314

AN EQUAL OPPORTUNITY EMPLOYER

ZUCO644 921276075 1A04 44 02/23/05
RETURN TO SENDER
:ZUCOTTO WIRELESS
MOVED LEFT NO ADDRESS
UNABLE TO FORWARD
RETURN TO SENDER